OPC UA More Than a Communication Protocol

Lauri Haapanen, Markus Sintonen, Andreas Frejborg
Neste Jacobs Oy, NAPCON Industrial ICT, PO Box 310, FI-06101 Porvoo, Finland
tel. +358 10 458 11, napconsuite@nestejacobs.com, www.nestejacobs.com

KEYWORDS OPC UA, Digitalization, Information Modelling, Automation

ABSTRACT

This paper describes how OPC UA provides rich view and flexible access to the industrial data. OPC UA server acts as a graph database which is well suited for creating highly versatile information models. This graph is based on nodes and relationships between them creating a network of nodes. The graph database has a flexible set of services for obtaining information about the structure of the graph. OPC UA standardizes the services for querying information from the contents of the server. These services allow querying the information from different perspectives. For example user can acquire hierarchical structure of the information model as well as to extract information from the graph as a whole. Also a subset of the graph can be queried based on a certain structure of the graph. This kind of graph like structure creates new possibilities for modelling and accessing the complex information sets that e.g. process industries commonly face. For example one could extract all of the pressure measurements from a certain type of process equipment based on a topological UA model of a plant. This brings new opportunities for creating intelligent software and services utilizing not only the process measurements but also the structural information contained inside the UA graph. UA information models bring structure to the otherwise unstructured and hard to interpreted data that traditional key-value pair automation databases offer. OPC UA also standardizes how the information is represented which allows easier interpretation of the data compared to traditional proprietary automation systems. Standards exists for simple measurements all the way up to complex devices for example analyzers and other industrial equipment. This makes it possible to create new services based on the plant information which has been previously impossible due to proprietary nature of the automation systems. - This brings digitalization to the plant level.

1 INTRODUCTION

OPC UA provides both standardized access to the industrial information as well as unified view to the data. The standard provides ways to model the industrial data so that it can be interpreted by any UA conforming application. This creates new possibilities from low level control application all the way up to high level reporting applications. The OPC UA information is exposed inside a server via a graph whose basic building blocks are nodes and references between the nodes. This combined with standardized information models such as those specified in OPC UA companion specifications creates quite a flexible way to express the industrial information. OPC UA also provides standardized access to the server graph via service interfaces that can be used to query single nodes or the graph contents as a whole. (1)

First this paper describes basic concepts of the OPC UA information modelling. Later on the paper shows few higher level example implementations and also discusses future developments. This paper focuses on the needs of the information management in process industry and how the OPC UA fits there.

2 OPC UA INFORMATION MODELLING

One of the strengths of the OPC UA is the ability to add semantics to the industrial information such as measurement data. In traditional automation information systems the measurement data is usually represented based on key-value pairing such that certain measurement position name identifies each of the measurements. Certainly these systems also have some metadata associated with the measurement position but generally there is no unified view to this additional information. Instead it is highly depending on the vendor specifics. OPC UA has ability to bring structure to the otherwise scattered data. In addition to this, the specification also specifies standardized ways for querying these structures. (1)

The basic building blocks of the OPC UA information models are nodes and references between them. These nodes and their relationships form a so called mesh network or a graph. In OPC UA servers these graphs are queryable and possibly dynamically changing. This type of a server can also be thought as a graph database. OPC UA also standardizes the service interfaces for the server queries which include basic read operations all the way up to complex chained queries. The picture below depicts that relationship between a graph, nodes and references. (2)

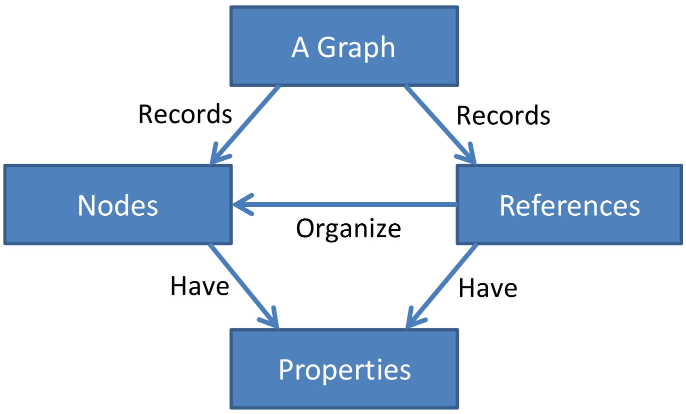

Figure 1. The general concept of a graph

2.1 THE GRAPH CONCEPT

Goals of the OPC UA are to create standardized ways to access and model widely varying domain information. In OPC UA the information is modelled inside a graph which creates quite a flexible environment to represent different kinds of domain information. This can be adapted from basic measurement information all the way up to higher level information representing for example whole production units. (1) In the recent years graph database applications have gained popularity due to richness and abundance of the domain data. The domains that contain rich and dynamic data relationships are well suited for the graph databases. For example social modelling and industrial environments usually are part of this domain. The traditional relational database management systems (RDBMS) highly rely on certain well predefined domain model. They are also highly depending on table joins which might become quite performance intensive with complex data. The graphs and databases containing these are optimized for highly connected data which for example process industry faces. (3)

The graph concept forms a basis for modelling the connected data and the OPC UA services provide standardized ways to query this information. This concept brings up the next problem which is information interpretation. Interpreting a graph containing nodes and references might be quite difficult from an application point of view that utilizes this information. The same problem that usually is faced with highly complex domain models inside RDBM systems. Fortunately OPC UA brings two level solution to this. These are standardized low level information representations and OPC UA companion specifications leveraging this to the higher level abstractions. (1)

At the lower level the graph is divided into namespaces which create a division between the standardized information representation and the vendor specific supplementary information. In OPC UA the graph is usually referred as an address space since each of the graph nodes are uniquely identifiable inside a namespace. This is also depicted in Figure 2. When querying the standardized namespaces the applications can be sure on what to expect about the graph contents. The references of the graph also have certain semantics related to them. At the low level they are divided into hierarchical and non-hierarchical references where the first one may not form circular relationships between nodes. These reference semantics allow applications to traverse the graph in a way that they can find the connected data they are looking for. Furthermore the specification also segments the graph into groups of nodes having a certain common structure. This is the OPC UA type definition concept where certain groups of nodes inherit a common well defined structure. This concept can be somewhat compared to the object oriented programming where classes define their instances. Also in OPC UA the types define their instances. Therefore the types are referred as type definitions in the specification. The standard also creates some base definitions for example measurement representations containing actual measurement value, ranges and units. Companion specifications make extensions based on the type system by creating models for different kinds of equipment and other resources. Of course the vendors can further extend these types by creating sub types of the standard types. Therefore the applications utilizing these types can be sure on what to expect from the node hierarchy when traversing the graph. (2) (4)

2.2 QUERY FEATURES OF THE GRAPH

OPC UA provides various interfaces for obtaining information about the contents of the graph inside the UA server. On the basic level the UA provides read service for reading data related to an individual node. In OPC UA each node has various attributes related to them. These are for example name, description and value of a node. The node value may also have historical data associated with it. The structure of the graph is usually queried using *Browse* and *Query* UA services. Here the *Browse* service provides information about the structure of the graph related to an individual node e.g. to get the connected nodes. The UA *Query* service is used to read the contents of a graph as a whole. This service allows simple queries for example to fetch all the nodes whose names start with a specific string. On the other end it also supports very complex queries that can filter nodes based on the structure of node and its hierarchy. For example one could query all the nodes of a specific type having a certain value in a directly connected node. These features allow rich access to the domain information. The figure below shows the conceptual difference between the namespaces and the UA address space. The picture also depicts the difference between *Browse* and *Query* UA services. (4)

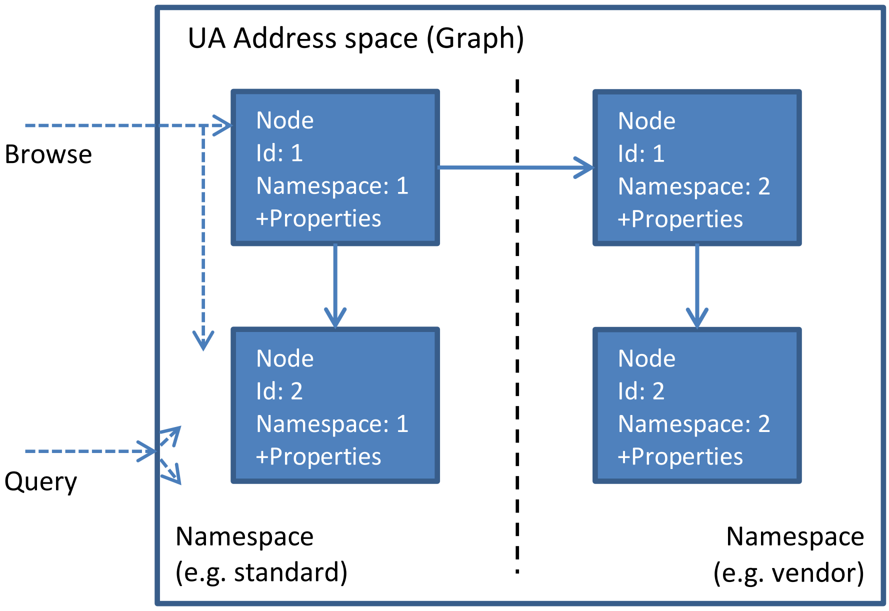

Figure 2. The division within the UA address space into the namespaces. Shows also the schematic difference between the *Browse* and *Query* services

In a typical process industrial application one is usually interested in higher level plant information and not just in an individual measurement positions. For example in some maintenance scenarios one could query all of the plant motor RPM measurements and compare them against each other. In this case all of the motors would have an association on a RPM measurement and possibly have some relationship to a plant process unit. In OPC UA the plant motors would have certain predefined type and the *Query* service could be used to fetch all of the motor instances inside the graph. The application would then explicitly know that the found motor instances also have the RPM measurement value associated with them. Further on the application could then fetch all of the historical RPM measurements from the last year which is also happening through the OPC UA protocol. In a typical scenario with the automation databases one would have to know all of the motor RPM measurement position names beforehand. There might be certain conventions on an organizational level on how the motor measurements have been named but still there are no guarantees that these are actually followed or even are up to date.

2.3 MUTABILITY OF THE GRAPH

The graph or so called address space of the UA server is highly configurable through service interfaces the protocol supports. These allow adding and removing of nodes as well as the same operations for references. The nodes are always created based on a specific UA type definition. Therefore they possibly inherit some predefined child hierarchy on the creation. Furthermore the created nodes always have one hierarchical parent node. This makes sure the graph is always connected. The additional relationships are created by the configurator using the reference creation service. (4)

The OPC UA type definitions control the basic structure of the graph since the created nodes always inherit the hierarchical structure of their UA type. The relationship between the created instances is somewhat user defined. There are also UA companion specifications that give rules about the created nodes relationships for certain domain data. For example ISA95 for OPC UA companion specification creates rules for creating node relationships based on the ISA95 standard. This standard specifies for example how relationships between industrial equipment and resources are modelled. On the other hand user might just simply configure the graph in a hierarchical tree where folder structure organizes the UA type instances. (5)

3 APPLICATIONS IN PROCESS INDUSTRY

In process industry the automation data is usually scattered around different automation systems possibly from different vendors. This creates a landscape where there are no common rules on how to find the specific data needed. Usually this requires some manual burden to find the different measurement data across automation systems. OPC UA information models and protocol can be used to create a mapping for the existing automation measurement data. The picture below depicts an example where pump measurements from distributed control system (DCS) server are mapped behind structured OPC UA information models. Structuring is here depicted inside a process information management system (PIMS) server. The picture only partly depicts the information model since the motor instances also have references to their type definitions. Also the motors have some hierarchical relationship to a parent node. Here it might be for example a process unit and further on a process area.

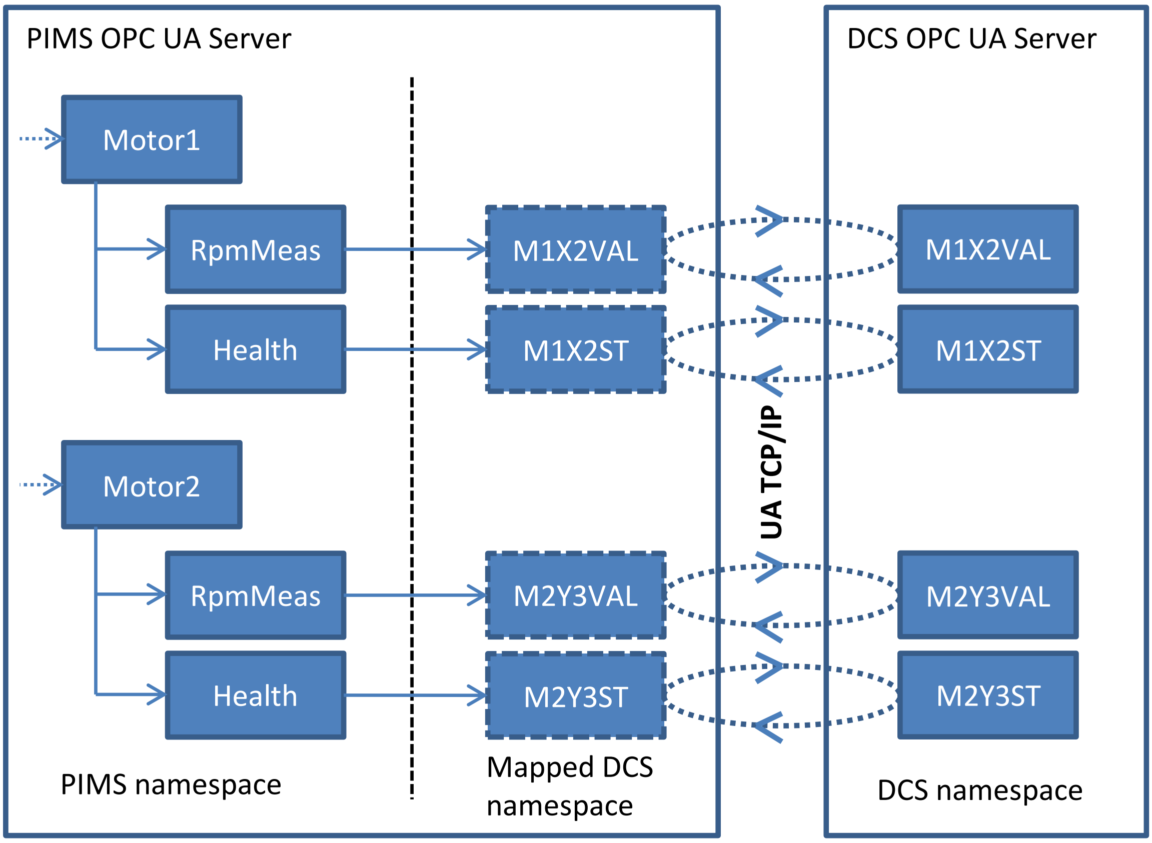

Figure 3. Example on the mapping of automation data behind UA information models

The picture also shows how for example *RpmMeas* node has a relationship to an automation node *M1X2VAL*. This reference might have a type of an *input* indicating that the value of the automation node is copied to the *RpmMeas* node. *RpmMeas* and *Health* nodes have child relationships to the parent motor instance node. It can be noted that the contents of the DCS server are mirrored to the PIMS UA server. This is called address space mapping or gateway functionality. This allows seamless access to the underlying system through the PIMS server e.g. all of the DCS server nodes are accessible through the PIMS server. Furthermore this allows creation of information model references between the different systems as the above picture depicts. The above model only depicts motors as an example but of course applications are much richer when combined with different types of process equipment from valves all the way up to complex analyzer devices. (2)

The applications that could benefit from this kind of information modelling range from control to reporting applications. The information is more available since it is behind standardized protocols and the structure of the information is well defined. This allows richer access to the plant information.

3.1 EXAMPLE IMPLEMENTATION

The OPC UA information modelling combined with gateway functions allows interesting possibilities to access the existing industrial information as shown in Figure 3. To allow this kind of functionality the OPC UA server product has to support various features that the UA protocol allows. These are the flexible graph modifications, gateway functions and query features as well as the security that the OPC UA provides. NAPCON Informer OPC UA database server developed by Neste Jacobs Oy is one example of an UA server implementation that allows these. (6) Furthermore it also supports historian features and the standard UA features that one would expect. This allows mapping of an existing DCS server behind information models providing standardized semantics to the industrial data. It has been used in various control applications where DCS data has been reorganized behind UA information models for these specific control applications. Another example use case is in energy reporting where existing automation sensor data was reorganized behind UA device integration companion specification (OPC UA DI). (7) Here the voltage and power measurement devices were then all queried from the UA graph and the energy report was based on this UA information as well as the historian data. This also allows flexible browsing of the sensor data from different perspectives. The maintenance of the application is also flexible since the report can be directly based on the UA information models so the actual report needs no updating. Only the UA graph needs device addition and the reference to the DCS source. This can be done using any UA compatible client program.

3.2 FUTURE DEVELOPMENTS

The challenge in process industrial information management is to find a common point of view to the automation data and plant semantics. There is an ongoing effort to create OPC UA companion specification based on Data Exchange in the Process Industry (DEXPI) standard. (8) This would allow combining the plant P&ID information with the automation measurement data. This combination of measurements and equipment information would bring true semantics to the process information. For example these semantics could be used in numerous applications that need information about e.g. which automation measurements are related to a specific distillation column.

Acceptance of the DEXPI standard is still unknown in real applications. Some of the engineering companies and plant developers still use legacy ways to store the plant schematics. In the worst case the schematics and P&IDs are just pictures with no way to interpret them programmatically. The industry as a whole requires some refinement in the work processes to bring digitalization to the plant level. In the end the customer who owns the plant must be aware enough to require the specific ways of working e.g. correct storage format of their P&ID data and OPC UA compatibility from the automation vendors. There is also need for easy to use tooling to manage e.g. the OPC UA graph databases in the shown scenarios. Also tooling for querying the UA databases based on plant schematic information must in place for the end users.

4 REFERENCES

1. **Mahnke, Wolfgang, Leitner, Stefan-Helmut and Damm, Matthias.** *OPC Unified Architecture.* s.l. : Springer-Verlag Berlin Heidelberg, 2009.

2. *OPC Unified Architecture Specification Part 3: Address Space Model Release 1.03.* **OPC Foundation.** 2015.

3. **Neo4j.** Relational Databases vs. Graph Databases: A Comparison. [Online] https://neo4j.com/developer/graph-db-vs-rdbms/.

4. *OPC Unified Architecture Specification Part 4: Services 1.03.* **OPC Foundation.** 2015.

5. *OPC Unified Architecture for ISA-95 Common Object Model Companion Specification Release 1.00.* **OPC Foundation.** 2013.

6. **Neste Jacobs Oy.** NAPCON Informer. [Online] http://www.napconsuite.com/products/napcon-informer/.

7. *OPC Unified Architecture for Devices Companion Specification Release 1.01.* **OPC Foundation.** 2013.

8. **DEXPI group.** Data EXchange in the Process Industry (DEXPI) group and OPC UA. [Online] 2016. https://www.automaatioseura.fi/site/assets/files/1552/nikolaos_papakonstantinou_opcdayfinland2016_opc-2016_dexpi-opcua.pdf.